BUZZ CHRONICLES > TECH
Saved by @Mollyycolllinss
See On Twitter

Twitter Thread by Sunil Kumar

Sunil Kumar
@sunilc_

Are you a Senior Developer & struggling to switch your job?

Here's a thread on how you should prepare before sending out job applications:

■■■

First of all know that there's a lot of demand for good software developers.

It's hard for companies to hire good mid level developers. Some of the interviewers I've talked to recently have mentioned this!

So, if you're good, you can always get into a high paying job.

I have 6.5 years of experience mainly as a Backend Developer. Although I do work a bit of front-end tasks as well as part of my current job.

In this thread I'm going to talk about how I would prepare if I was going to apply for a job interview.

As a mid level developer having 4 - 8 years of experience, you will be expected to have certain skills to be hired by companies. Skills like:

- Problem Solving
- HLD
- LLD
- Mentoring / Code reviews
- Taking responsibility
- Leading end-end development & release of projects etc

Problem Solving:

Even though we mostly never write a lot of game changing Algorithms and use data structures, you're still expected to know the most common ones and how to use them to solve problems:

- Tree
- Linked List
- Heap
- Hash Map
- Array etc

You should practice solving Algo & DS questions which are typically covered in the first round of your interviews.

Leetcode is a good choice and covers most of the common questions and the concepts one needs to know.

Practice for about 2 months to solve around 100 questions in Medium and Hard categories.

You can start with Easy questions. But that's just to get into the zone since you might not have solved these questions in years.

But focus on Medium and Hard questions on Leetcode.

A very important point to note is that don't try to remember solutions to questions. It's never gonna help you. If you're doing that, you're just wasting your time.

Instead focus on learning the most common techniques used to solve Algo & DS questions.

https://t.co/IfUEwcDuCN

> Interview Tip:
>
> Common techniques used to solve Algo / DS problems:
>
> - use a hash map
> - sort the array first
> - use sliding window
> - use DP
> - use two pointers
> - binary search
> - backtracking
> - bit manipulation
> - use array elements as index
>
> (1/2)
>
> — Sunil Kumar (@sunilc_) December 12, 2020

When preparing for this round, practice writing code for each of the solution in any languages of your choice.

Just knowing how to solve a problem is not enough, you should be able to write bug free code in the given time.

This round gives a good idea about your coding skills.

System Design:

System Designing is another important skill you will be evaluated on. This includes both LLD and HLD.

LLD -> Low Level Design
HLD -> High Level Design

Low Level Design:

Typically LLD rounds focus on low level components of a system which covers below skills:

- class design
- design patterns
- OOP principles
- SOLID principles etc

Here's a reference:
https://t.co/lg4z7gPCy0

> Different things asked during your LLD (Low Level Design) interview round:
>
> - use case diagram
> - activity diagram
> - class design
> - database design
> - code / pseudo code
> - algorithm used for the core logic
>
> — Sunil Kumar (@sunilc_) January 14, 2021

Here's a great course to prepare for LLD round "Grokking the Object Oriented Design Interview":

https://t.co/COPTtHXPCQ

High Level Design:

In a HLD round, you'll be given a problem statement and you're expected to design the high level components involved in the system. You need to take care of different use cases which cover

- fault tolerance
- high availability
- data flow
- downtime etc

Here's pretty famous course to prepare for HLD round: "Grokking the System Design Interview"

https://t.co/Su63FPMFxH

Managerial / Culture Fit:

Another important round is either a Managerial / Culture fit round. Which typically involves discussions around your previous projects, challenges you faced and other standard questions like:

- Most challenging problem
- Strengths / Weaknesses etc

To prepare for this round you should know your current project in and out.

You should know about all the different corner cases handled in important features, the design decisions taken, the services chosen etc.

For Example you should know:

- why a SQL database is used in your service?
- why a redis cache is used?
- why a SQS queue is used?
- how a particular error case is handled ?
- when an internal goes down how is scenario X handled?

I cannot emphasise enough on knowing your current project thoroughly.

If you're able to justify all the questions asked on your current project, the interviewer will be impressed & will know that you're actively contributing in the project.

Last but not the least, be prepared to ask right questions at the end of every interview. Asking right questions shows that:

- you're genuinely interested in the company
- gives a good understanding of their tech stack, work, work culture, policies etc
- helps to make a decision

Here's a complete guide (by @RandallKanna) for developers on Job Hunting, Acing the Interviews & Landing the job:

https://t.co/JmG4SKCYYQ

If you like this thread please retweet the first tweet to help me reach more people.

Download my free e-book which contains all of my viral content from the past 8 months here: https://t.co/xCLUcGqZMF